

# Why Study Large Scale Computing (aka Mainframes)

# What is “Large Scale Computing”?

- Developing “really big” applications
- Focus on highly available, transaction processing systems
- Using the “mainframe” as a server

# Capabilities of a Mainframe

- Large-scale transaction processing  
(thousands of transactions per second)
- Support thousands of users and application programs
- Simultaneously accessing resources
- Terabytes of information in databases
- Large-bandwidth communications

# An Example

A Large System in a large Company

# JP Morgan Chase Overview

## **By The Numbers...**

- Net Revenue: \$99B
- Net Income: \$14.4B
- Assets: \$1.2T
- Headcount: 174,360
- Technology & Communication Expense: \$3.5+B

# JP Morgan Chase Overview

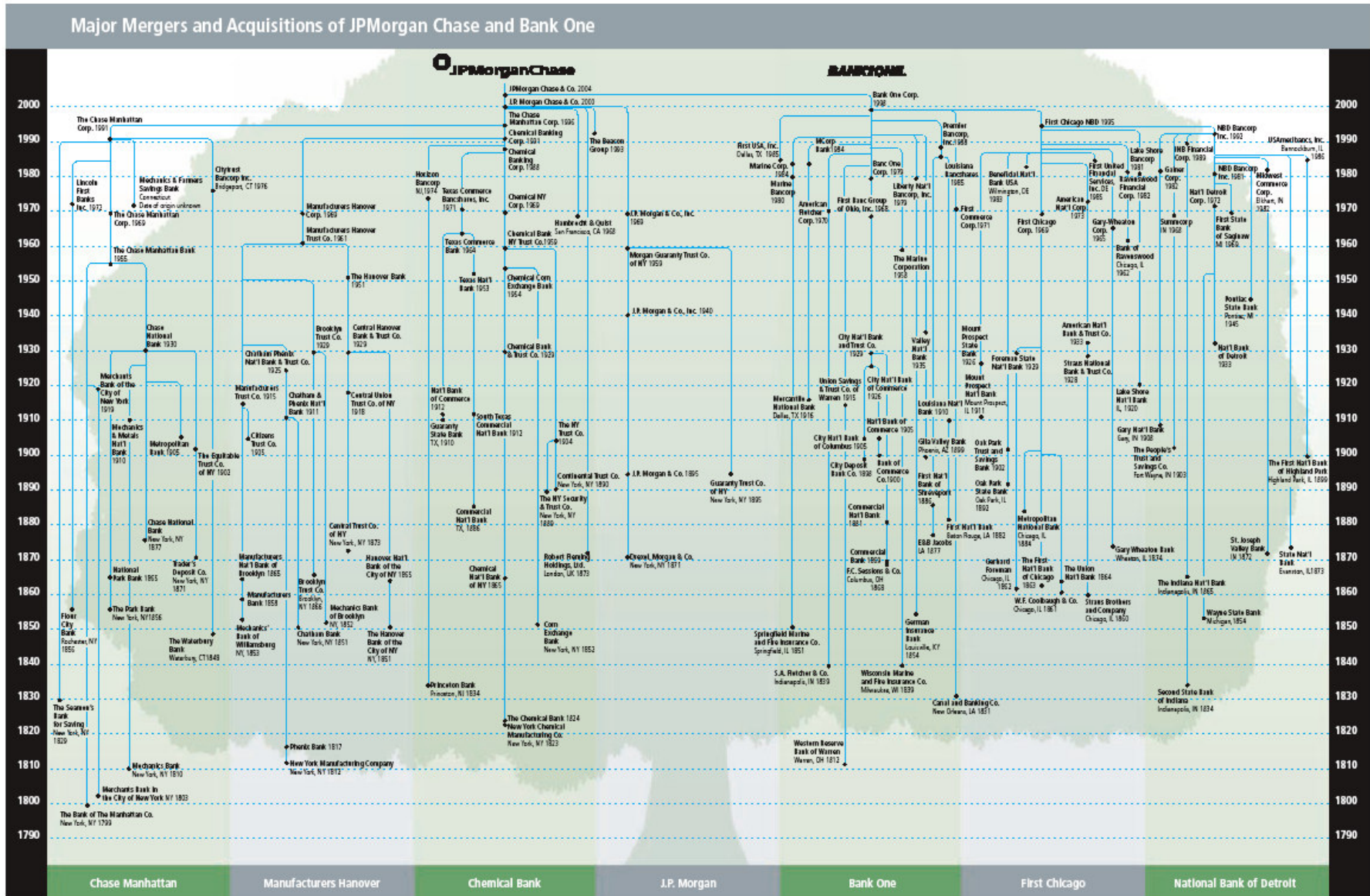
## **JP Morgan:** Wholesale banking

- Relationships with over 99% of the Fortune 1000 companies
- \$3.2 trillion in wire transfers daily

## **Chase:** Consumer & commercial banking

- 2,600+ branches (& growing via acquisition)
- 110+ million credit cards in circulation

# The JP Morgan Chase “Merger” Tree



# Integration Challenges in Financial Services

## Key Challenges

- *Application integration challenges inherent from M&A efforts*
- Variety of applications and platforms across business units
- Business unit silos not conducive to customer integration

## Implications on Business & Technology

- Need to maintain a holistic view of the customer relationship
- Ensure a consistent understanding of the customer – across all businesses
- Distribute information within and across business units in an easy, reliable and cost effective manner.



# Technologies used within Banking

## Technologies used

- ✓ PCs
- ✓ Distributed (aka Unix)
- ✓ Large Scale Computing Platforms (aka Mainframes)

## Why mainframes?

- Large-scale transaction processing (thousands of transactions per second)
- Support thousands of users and application programs
- Simultaneously accessing resources
- Terabytes of information in databases
- Large-bandwidth communications

# Thoughts on Outsourcing

- Need to control strategic systems & processes
- JP Morgan Chase has “insourced” more than “outsourced” over the past several years.
- JP Morgan Chase has offices in over 50 countries – we live and work in a global marketplace
- Most companies outsource within the United States

“The issue boils down to how much of a company's IT should be outsourced. If you're J.P. Morgan, the answer is: almost none of it.”

Business Week (Oct, 2004)

# Why Insource?

## **Become the Low Cost Provider / Drive Down TCO**

- Reduce TCO through consolidation of existing internal applications
- Streamline/re-engineer business processes

## **Support proprietary initiatives**

- Rapid deployment - control prioritization of development efforts
- End-to-end ownership enables protection of proprietary info

## **Control Our Architecture**

- Improve service levels (predictability, simplicity, future growth) by streamlining end-to-end systems and processes

# An example of a Large System

# The Credit Card Core Processing System

## **What does a “core processing system” do?**

- Authorizations (your card “swipe”) - Credit & Fraud analysis
- Statements - transaction history, rewards
- Customer & Account System of record
- Interact with Visa & Mastercard

## **The Size of a “Large” Chase System**

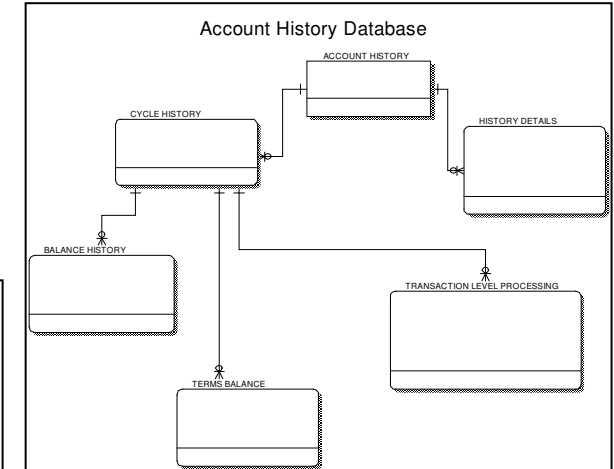
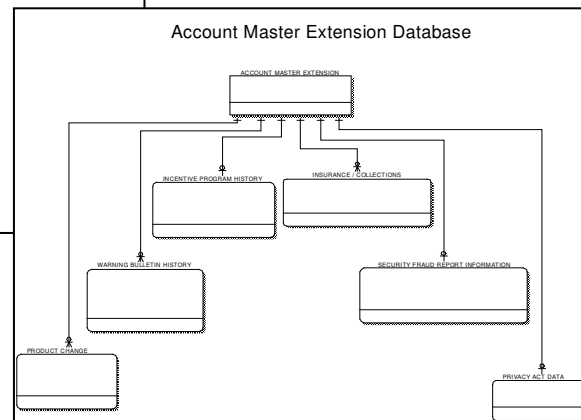
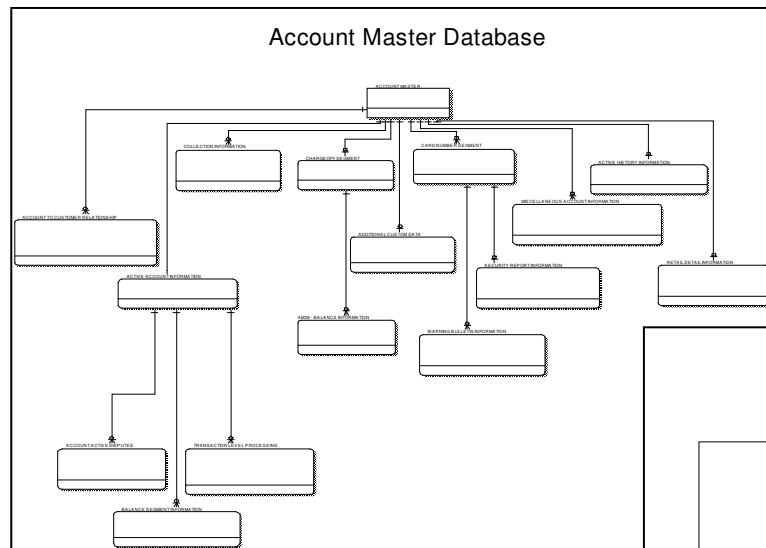
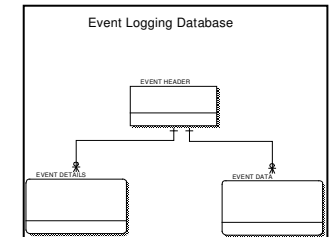
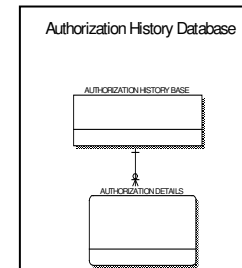
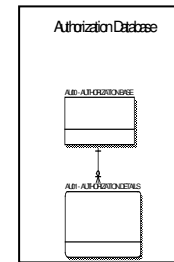
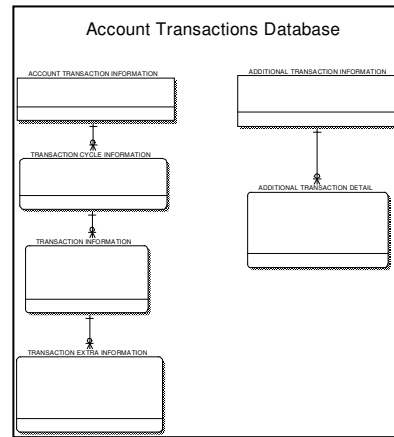
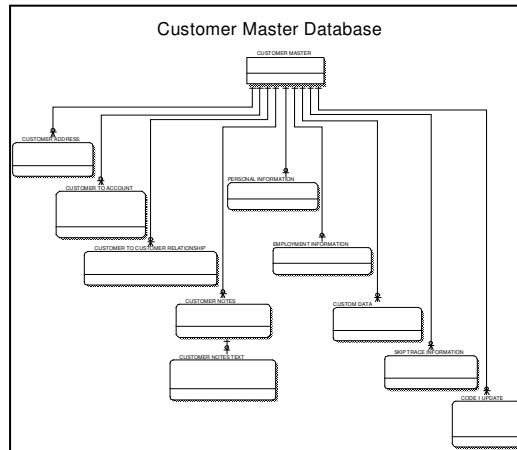
- 68K Source Code Components
- 10K+ COBOL Executables
- 12K+ batch jobs
- 50TB of data (5,000 Production Database Areas)
- 100M Accounts
- 15M POS decisions a day (peak of 400/sec, each taking 100MS)

# The Key Characteristics of a Card Processing System

## Architecture

- IMS & DB2 Databases
- Separate Customer and Account structure
- Option (business) driven
- Highly available, 24x7
- Information accessible via: Native 3270, MQ and XML
- Workflow management to streamline customer service
- Parallel nightly batch processing
- Real-time credit & risk evaluation (in less than 200 milliseconds)

# Data Model for a core processing system



# Data for a core processing system

Entity Name	Entity Definition / Description
<b>ACCOUNT HISTORY</b>	Account history root segment
CYCLE HISTORY	This is current cycle information that is rolled into history. A new history segment gets created for each new cycle.
HISTORY DETAILS	History Details of the three types of "Actions" or "Maintenance" - Promise to Pay, Credit Limit and Dispute, performed on an account (e.g. raise Credit Limit, Accept a promise to pay or put a transaction into dispute). A new segment is created whenever t
BALANCE HISTORY	This is the history from the balance segment. At Cycle, a copy of the balance segment is rolled into this segment, creating a segment for each transaction category (TCAT).
TERMS BALANCE	This is history of the balance history rolled up to the terms balance (TBAL) level. The multiple TCATs can roll up into one or more TBALs.
TRANSACTION LEVEL PROCESSING	This is history from the transaction level processing. This is for money that is either Deferred or Same as Cash. An example of "Deferred " is no interest, no payment for 6 months. "Same as Cash" is payment only, no interest.
<b>ACCOUNT MASTER</b>	Account Master root segment. This is the largest IMS segment for account. This segment stores over 80 Account Level fields that provide information about the status of the account. For example, Past Due, Activity today, Fraud block, etc.
ACCOUNT TO CUSTOMER RELATIONSHIP	The purpose of this segment is to link the Card Number to the Customer ID. The segment contains Card Level information, one for every customer, regardless of whether or not the card numbers are the same or different.
ACTIVE ACCOUNT INFORMATION	This contains active account information. It provides a picture of the total balance on the account. It contains past due, over the limit information and many date fields such as date of last purchase, date of last cash, date of last debit, date of last
ACCOUNT ACTIVE DISPUTES	This is the active dispute segment. There is one segment for each transaction in dispute.
BALANCE SEGMENT INFORMATION	This is the balance segment. There is one segment for each TCAT that contains money (Examples: Fee Money, Purchase Money and Cash).
TRANSACTION LEVEL PROCESSING	This segment records information on "Same as Cash" or "Deferred" transactions. There is a segment for each "Same as Cash" or "Deferred" transactions.
COLLECTION INFORMATION	This segment is redefined in three ways to contain the three types of collection information: Past Due, Over the Limit and Promise to Pay.
CHARGE OFF SEGMENT	This segment is generated when an account charges off. It contains information about the chargeoff. There is no occurrence prior to an account charging off.
BALANCE INFORMATION	This segment contains the balance information at the time of the account charge off.
CARD NUMBER SEGMENT	This segment contains Fraud and Warning Bulleting information. There is one segment for every card number. This is the parent segment for Warning Bulletin Information and Security Report Information.



# Data for a core processing system

Entity Name	Entity Definition / Description
WARNING BULLETIN INFORMATION	This is Warning Bulletin Information. This is information about blocking the account at the association level.
SECURITY REPORT INFORMATION	This is security report information and it can only occur once. It contains information collected from a customer for incidents such as lost / stolen card.
ACTIVE HISTORY INFORMATION	This is active history segment.
CRITICAL CUSTOMER DATA	Critical Customer Information. This segment was created for performance reasons to enable storing Customer Information in the Account Database so that batch processes do not have to access the Customer Database.
ACCOUNT MASTER EXTENSION	Account Master Extension root segment. It contains account data that is stored separately to facilitate easy access. The segment was created for performance reasons because the Account Master Database already contains a lot of data.
PRODUCT CHANGE	This segment is used when the Customer trades from one product to another product (e.g. Master Card to Visa Card to AAA card to Disney Card.) It records a snapshot of many account level fields at the time of the product change.
WARNING BULLETIN HISTORY	This is an extension record that contains additional data related to the Warning Bulletin contained.
INCENTIVE PROGRAM HISTORY	Consumer Rebate/Reward segment, one occurrence at Cycle. This contains a history of incentive programs such as Consumer Rebate and Consumer Reward.
EXT. SECURITY FRAUD REPORT INFORMATION	This is additional Fraud Security Information.
PRIVACY ACT DATA	This is where privacy and opt-out information is stored.
<b>AUTHORIZATION BASE</b>	This is the authorization root segment. It contains authorization information such as total authorization performed on the account (number approved today, number declined today etc.).
AUTHORIZATION DETAILS	This is a child segment that is created for every authorization. There is one created for each authorization. The purpose is to withhold money from the available credit until the transaction posts. After the transaction posts, this segment is deleted.
AUTHORIZATION HISTORY BASE	This is the authorization history root segment. It contains authorization information such as total authorization performed on the account (number approved today, number declined today etc.).
ACCOUNT CROSS REFERENCE	This segment ties together the keys of several databases so that if one key is known, the rest can be looked up. It ties the application number, customer number and rewards number together.
ACCOUNT TRANSACTION INFORMATION	This is the root segment for the account transaction database.
TRANSACTION CYCLE INFORMATION	This segments contains information that delineates the beginning of the transaction cycle. The key is "Date Statement Begin".
TRANSACTION INFORMATION	This segment contains the transactions for the cycle.
TRANSACTION EXTRA INFORMATION	This is a child segment of the Transaction Information segment.
ADDITIONAL TRANSACTION DETAIL	Additional transaction detail segment containing SKU type information on retail transactions, including Department ID, SKU level description, item quantity, item price and SKU or tracking number.

# Data for a core processing system

Entity Name	Entity Definition / Description
<b>CUSTOMER MASTER</b>	Customer database root segment. There is one segment for each customer on an account. The keys are Client Number and Customer ID. Customer Name, SSN, Zip and Phone numbers are also stored and serve as secondary keys for lookup.
NAME LOOKUP	This is the Customer Cross-reference segment used for searching (using the ICAL screen) on Customer information (search by Customer Name) to locate specific accounts. This is also regarded as a secondary index.
SSN LOOKUP	This is the Customer Cross-reference segment used for searching (using the ICAL screen) on Customer information (search by SSN) to locate specific accounts. This is also regarded as a secondary index.
ZIP LOOKUP	This is the Customer Cross-reference database used for searching (using the ICAL screen) on Customer information (search by Zip) to locate specific accounts. This is also regarded as a secondary index.
PHONE LOOKUP	This is the Customer Cross-reference database used for searching (using the ICAL screen) on Customer information (search by Phone number) to locate specific accounts. This is also regarded as a secondary index.
CUSTOMER ADDRESS	This is the Customer Address segment.
CUSTOMER TO ACCOUNT	This establishes the relationship between the customer and specific account numbers. Since the account number is part of the key, there is a different segment for each account number.
CUSTOMER TO CUSTOMER RELATIONSHIP	Establishes the relationship between the primary Customer and additional Customers. For example, Father, Daughter etc
CUSTOMER NOTES	This is a Customer notes segment.
CUSTOMER NOTES TEXT	This is a child of the Customer Notes segment and it contains the details of the notes.
PERSONAL INFORMATION	This segment contains Customer Specific personal information such as hobbies, marital status, checking and savings account numbers, widowed etc.
EMPLOYMENT INFORMATION	This segment contains employment information.
CUSTOM DATA	This is Customer custom data segment.
SKIP TRACE INFORMATION	This is Collections skip trace segment. This is used to share information with other Credit Card Companies for Collection or recovery purposes with regards to delinquent accounts.
<b>EVENT HEADER</b>	Event root segment.
EVENT DETAILS	This data is used for Batch Event Reporting.
EVENT DATA	This segment contains data for on-line viewing of events, memos and memory messages. There are rules that determine how long the information is kept, based on the event type.
<b>SCORE</b>	This is the scores root segment.
SCORE DATA	This is the scores segment that contains detail information for each score type. The scores within this segment can be used within risk calculations.
<b>BUREAU SCORE</b>	This contains supplemental information for a bureau score. It contains some limited bureau information, such as number of disputed trade lines, trade lines and public records. A trade line is an entry in the credit bureau record for a customer.

# Data for a core processing system

Entity Name	Entity Definition / Description
<b>PURCHASE ANALYSIS</b>	Purchase analysis root segment.
ACCOUNT PURCHASES	Account level purchase analysis details for general purchases and cash only, life to date, cycle to date and prior year.
PURCHASES BY MERCHANT	Purchase analysis details by transaction summary code which is selected in the purchase analysis option. For example car rental, medical expenses. Each category will generate one of these segments.
<b>APPLICATION ENTRY</b>	This is the Application Entry Root Segment. This segment is the first IMS segment to be created when an application gets into new account analysis. It contains a number of application information that eventually goes to the account master segment
APPLICATION CUSTOMER INFORMATION	This is the Application Customer Information segment.
APPLICATION AUTO DECISION ACTION	Application Auto Decision Action segment. This segment is used for New account analysis internal processing, for example queuing, generating correspondence etc.
APPLICATION DATA ELEMENT VALUE DATA	Application Data Element Value Data segment. This is a client defined data element segment. The client defined data element aid the application of business rules, for example assigning scores for having a checking account.
APPLICATION MEMOS	This segment stores Application Memos. These are the memos generated during the processing of the application.
APPLICATION STATUS DATA	This is the application status data segment that stores the status of the application.
MESSAGE DATA	This is the application Message Data segment.
APPLICATION ALERT HITS	This is the Alert Hit Data segment. The alert file is a fraud database. If an application comes in that has an entry in the alert file (e.g. phone number, address), we will get an alert hit.
ALERT INFORMATION	This is the alert file root segment. This stores the data in the alert file.
ALERT ACTIVITY INFORMATION	This is an individual alert file segment either captured by the system or entered by an operator. It contains information about individual fraudulent names, addresses, phone numbers, social security number etc.
PIN MAILER KEYS	Embossing file segment for PIN mailer key information for individual records on the embossing database.
STAGE OF CARD REQUEST	Additional card request processing information on the embossing database.
EMBOSSING DATABASE	This is the embossing database root segment. Any request for cards, PIN, mailer etc are stored in child segments to this segment.
PIN MAILER	This is the PIN / PIN mailer segment. This stores information related to the PIN.
CARD MAILER	Embossing database segment containing card mailer information.
CMS CROSS-REFERENCE INFORMATION	Embossing database cross-reference record. This cross-references the individual Application Number to the embossing file transmission ID.
CARD INFORMATION	Embossing database segment containing card, formatting, embossing and magstripe encoding information.
INQUIRY LOGGING	Inquiry database logging root segment. Every operator ID defined will have an inquiry logging segment.
INQUIRY LOG DETAILS	This segment is created whenever an operator looks at an account level green screen. It contains the timestamp, operator screen looked at, what account, etc.

# Course Details

# Viability of the Mainframe

## Who uses mainframes?

- Most Fortune 1000 companies use a mainframe environment
- 60% of all data available on the Internet is stored on mainframes
- There are more CICS transactions processed than Web pages served

## Mainframe Resources:

- A growing issue for **mainframe** enterprises is the shrinking set of skilled staff to maintain the environment - Gartner 2004/2005.
- **Wanted: 20,000 IBM Mainframe Experts** - InformationWeek, Aug 2005

## Financial Impact to IBM:

Mainframe-related sales likely contribute around 25 percent of IBM's revenue and, according to some, close to 45 percent of their operating profit.

# Factors contributing to mainframe use

- RAS -- *reliability, availability, serviceability*
- Security
- Scalability
- Centralized control
- Workload management
- Partitioning / virtualization
- Continuing compatibility
- Evolving architecture
- Application enablement, complexity, variety
- Potential for thousands of users

# *The World Depends on Mainframes*











































Mainframes are a part of everyday life. When you:

<i>Turn on a light</i>	<i>Get a business loan</i>
<i>Make a telephone call</i>	<i>Process accounting records</i>
<i>Use your ATM card</i>	<i>Control inventories</i>
<i>Put money in a bank</i>	<i>Process payroll</i>
<i>Rent a car</i>	<i>Update personnel records</i>
<i>Purchase life insurance</i>	<i>Control an assembly line</i>
<i>Travel</i>	<i>Control a railroad</i>
<i>Send a package</i>	<i>Use corporate data bases</i>
<i>Track in-transit packages</i>	<i>Run a government agency</i>
<i>Trade stocks</i>	<i>Conduct international business/banking</i>

**And more...**

**... you are likely using a mainframe!**

# What Would You Title This Chart?

	IBM zSeries Z/OS	Fujitsu Primepower Solaris	Sun 15000 Solaris	HP Superdome HP-UX	IBM p690 AIX5L	Dell Intel SHV W2K	Unisys ES7000 W2K
Processor Technology							
System Performance							
Parallel Clustering							
Unplanned Downtime (S)							
Unplanned Downtime (C)							
Disaster Tolerance							

Based on  
September 2003  
Updated Model

Best



Worst

Gartner



# Course Objectives

- Understand key concepts required for large scale application development - which is independent of “mainframe” vs “distributed computing”
- Describe programs from an “Application Architecture” perspective
- Gain some real world insight into the application of “Computer Science” concepts

# Course Objectives

Be able to have an intelligent conversation with respect to Large Scale Computing

- Understand the differences when creating applications to run on a PC, traditional web-based applications and “really big” applications.
- Become “fluent” in the TLAs/Buzzwords
- Understand Hype verses reality



# Course Topics

1. System Architecture overview
2. Introduction to COBOL and JCL
3. Data: VSAM and Sequential Files
4. Application Architecture of transaction processing systems
5. Databases: **Relational and hierarchical databases**
6. Using the mainframe as a “server” (MQ, HTTP, J2EE)
7. Transaction Processing – **CICS**
8. High Availability - **Parallel Sysplex**
9. Security – **system security & security processes**

# Why Learn Cobol

- Most mainframe programs are written in Cobol
- Most applications are large  
(ex. 10,000,000 lines of code)
- Cobol can be very efficient for certain types of applications

# Course Format

- **Lectures** - Review key concepts - includes both applied (ex. compile/link a program) and foundational topics (ex. security)
- **Labs** - Focus on getting familiar with the mainframe environment
- **Online** - helping to create/use our “Wiki”  
([largescalecomputing.wikispaces.com](http://largescalecomputing.wikispaces.com))

# Our Wiki

- **Areas of the Wiki**
  - Course info (ex. reading assignments, schedule, ...)
  - General mainframe computing info (ex. terms & abbreviations,
- One contributes to the Wiki by **editing (improving) the web pages**, and as appropriate, contributing to the **discussion on specific web pages**
- It is expected that students will be an active reader / contributor to our web site!

# Our Wiki

The screenshot shows a Windows Internet Explorer browser window displaying the LargeScaleComputing Wiki home page. The browser's address bar shows the URL: <https://largscalecomputing.wikispaces.com/?responseToken=9ec6b2360bb0c71819045d4730e07093>. The browser's menu bar includes File, Edit, View, Favorites, Tools, and Help. The toolbar shows various icons for navigation and editing. The page title is "LargeScaleComputing" with a small tree icon. Below the title, there are tabs for "home", "page", "discussion", "history", and "notify me". The "home" tab is selected. On the left side, there is a sidebar with links for "New Page", "Recent Changes", "Manage Space", and a search box. Below these are links for "CSCI 388-Spring 08", "Overview", "Syllabus", "Lab 1", "Lab 2", "Lab 3", "Lab Discussion", and "Wiki Update". Under "General Topics", there are links for "Tutorials/References", "Buzzwords", "VSAM", "ISFP", "ISFP Editor", and "Java vs Cobol". At the bottom of the sidebar, there are links for "MF Capabilities", "Databases", "Transactions", "Security", and "Reliability". The main content area has a heading "Welcome to the Large Scale Computing Wiki home page." with an "Edit This Page" button. Below the heading, there is a paragraph: "This web site has information on CSCI388 as well as general topics on mainframe computing." followed by another paragraph: "These *general topics* contain useful information on large scale (mainframe) computing, that will be used (and updated) within the course, but the information is not specific to our class. Some of the general topics include:". This is followed by a bulleted list: 

- [Buzzwords](#) - A quick look up of acronyms often used in mainframe computing
- [Useful References](#) and tutorials - Articles and books (typically in PDF format)

Below the list, there is a section titled "Class announcements:" with the text: "Added how to install the [3270 terminal emulator](#) (for using on a home PC)". Another section titled "Adding Content to this web site" contains the text: "Almost all pages in this web site are editable (click on the edit button in the upper right part of the web page). Everyone is encouraged to add content to the web site (some of assignments will require you to update some of these web pages). Feel free to use this [test page](#) to learn how to use this Wiki space (there is also [help](#) if needed).". At the bottom, there is a note: "Note that in addition to update/creating web pages, you can (and should) be reviewing the "discussion" tab each of the pages (where you can *post* comments and *reply* to others that have posted comments).".

# Course Grading

- **5 % Lab 1**
- **10% Lab 2**
- **20% Lab 3**
- **20% Exams (two, combined they count 20%)**
- **5% Lab presentation**
- **10% Wiki update assignment**
- **20% Final Exam**
- **10% Participation (Wiki & Classroom)**

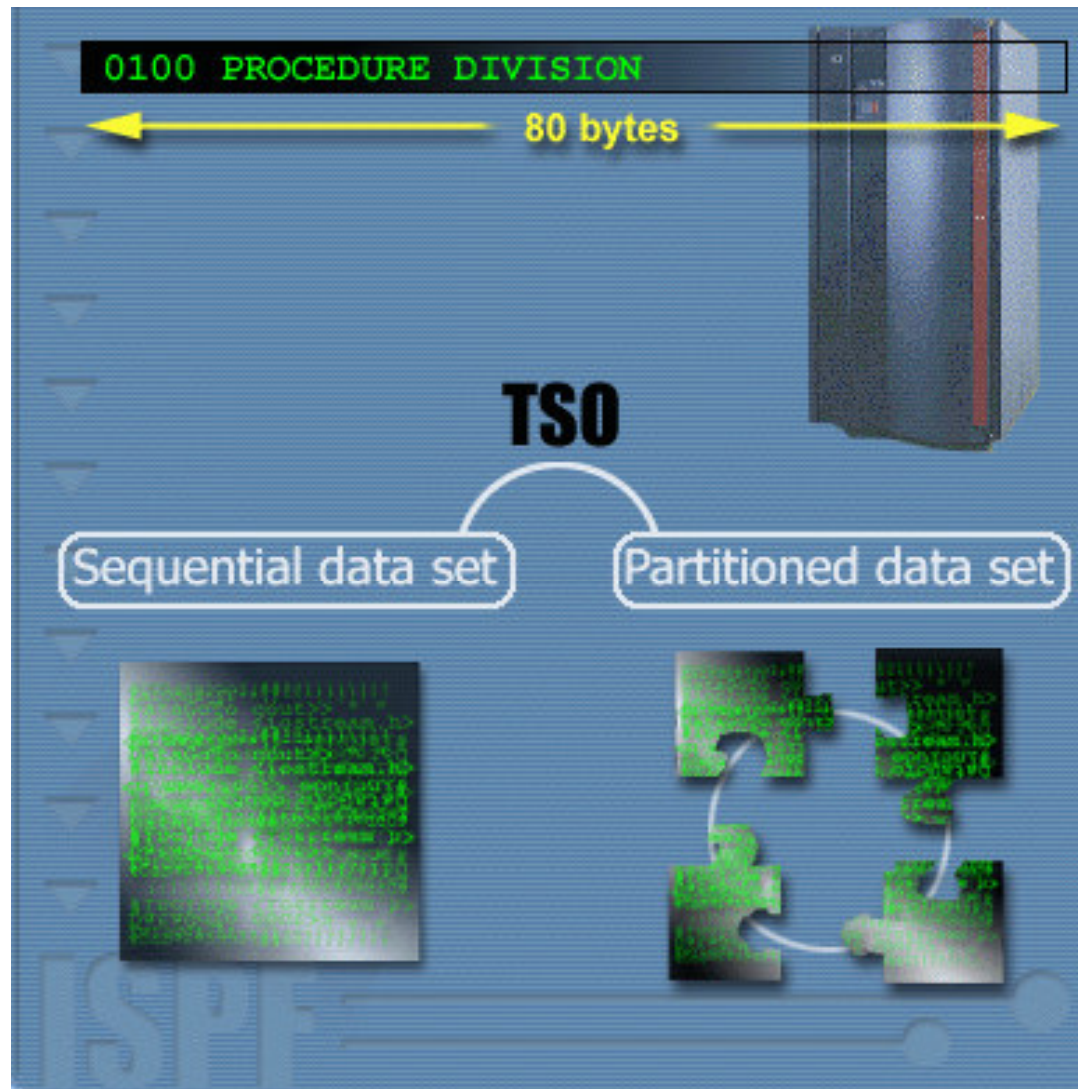


# Data Sets

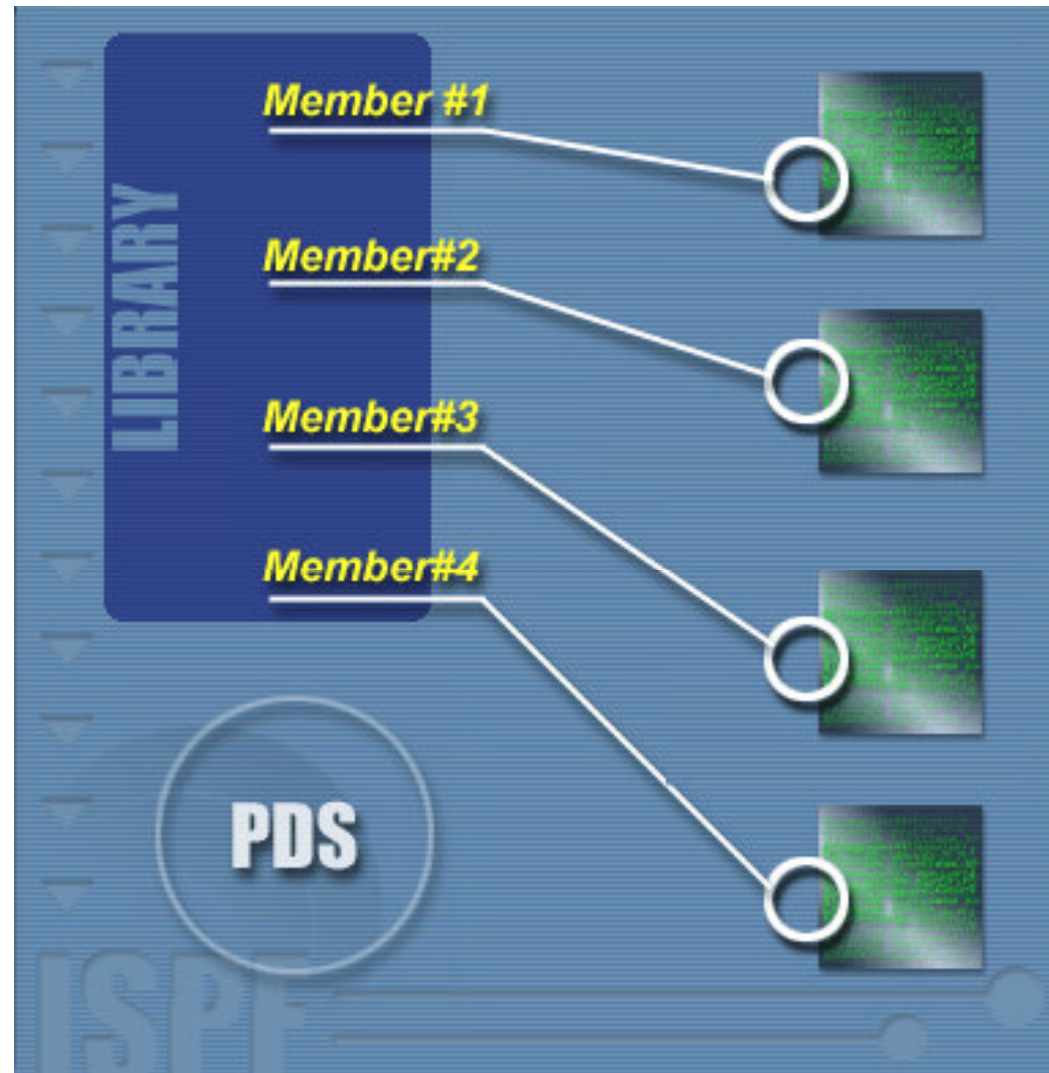
# What is a data set?

- **z/OS files are called data sets.** Before you can write data into them, space for data sets must be reserved on disk. The user is involved in specifying the amount of space as well as the formatting of it.
  - A data set is a collection of logically related data records stored on one disk storage volume or a set of volumes.
  - To use a data set, you first *allocate* it.
  - A data set can be:
    - a source program
    - a library of macros
    - a file of data records used by a processing program.

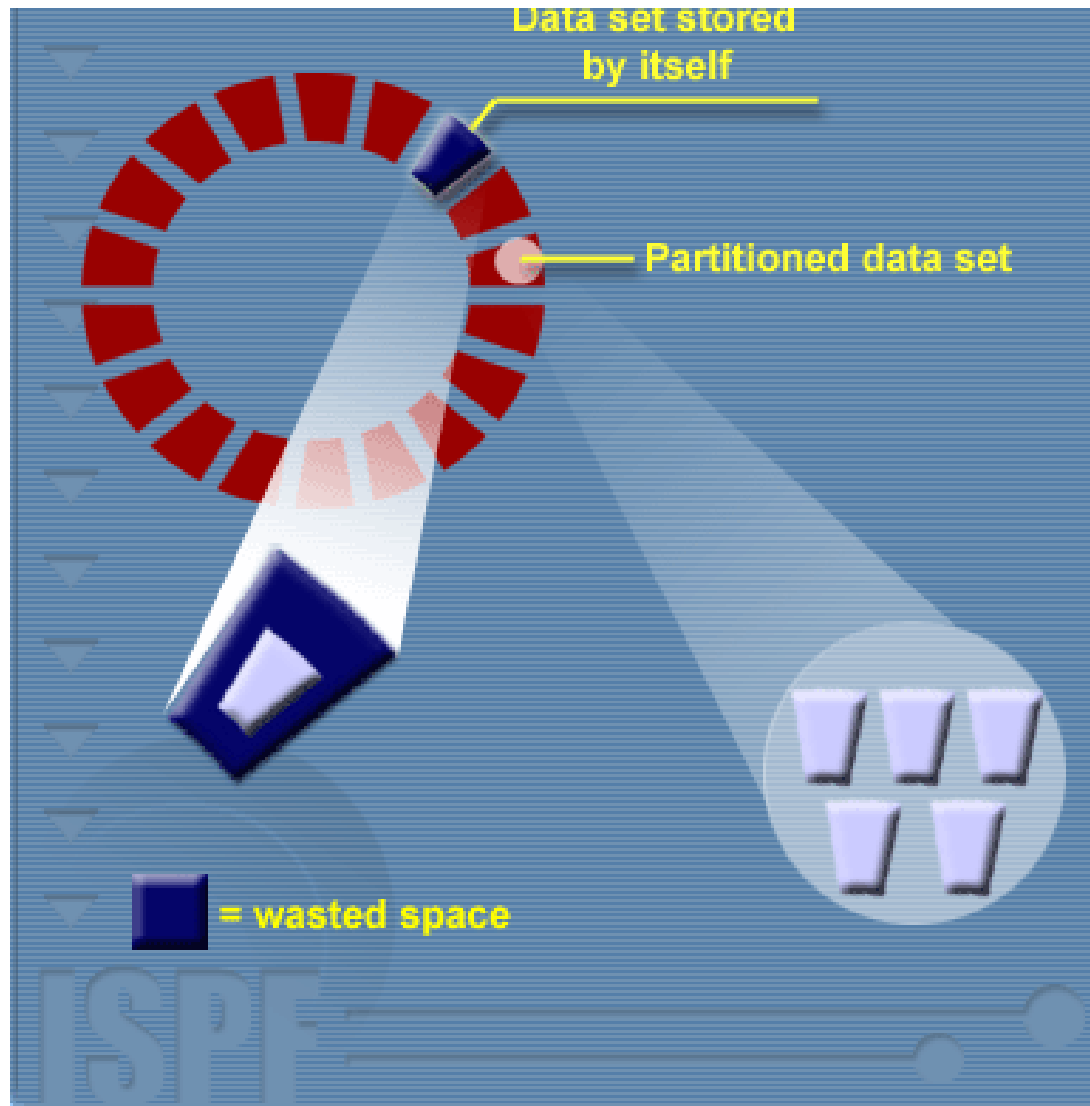
# Types of Datasets



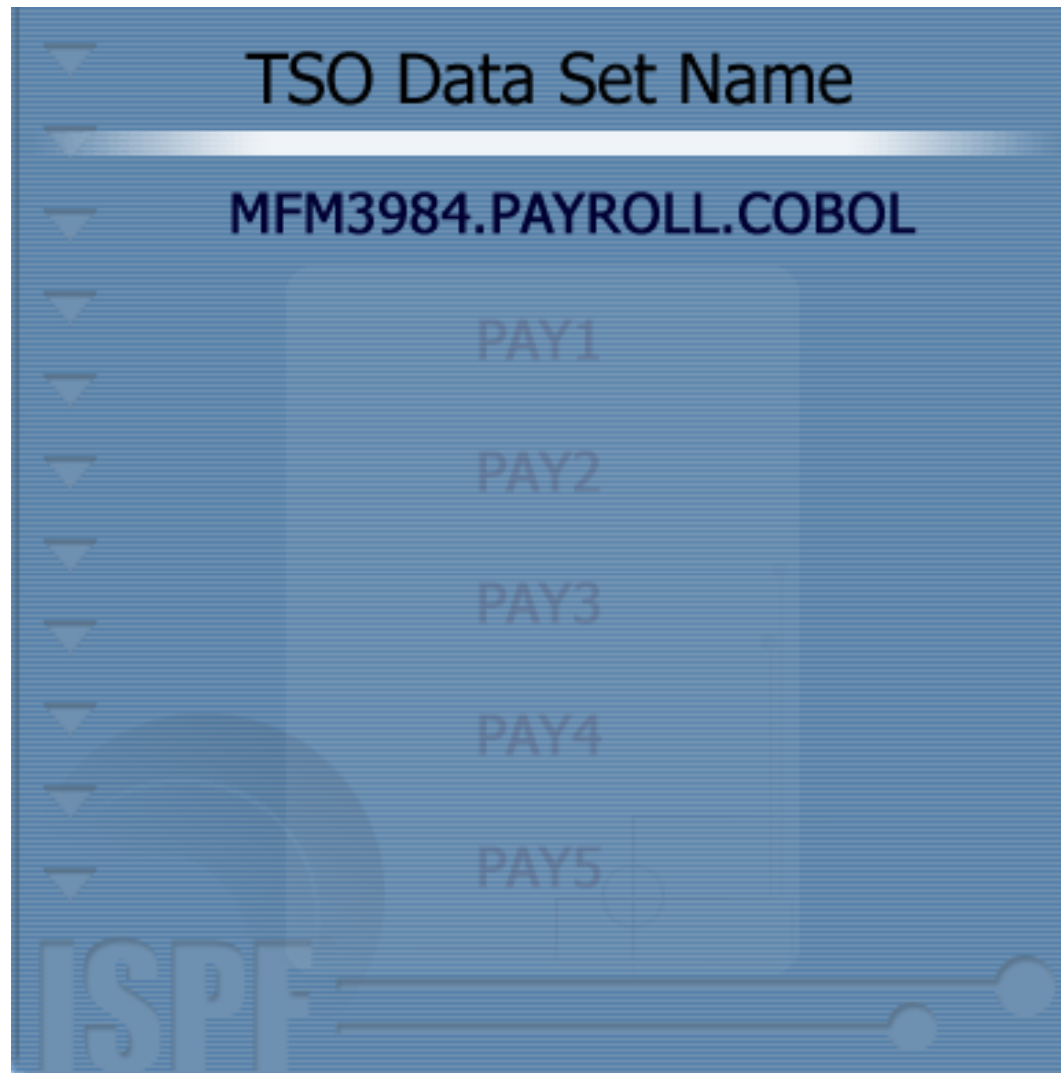
# Partitioned Data Sets (PDS)



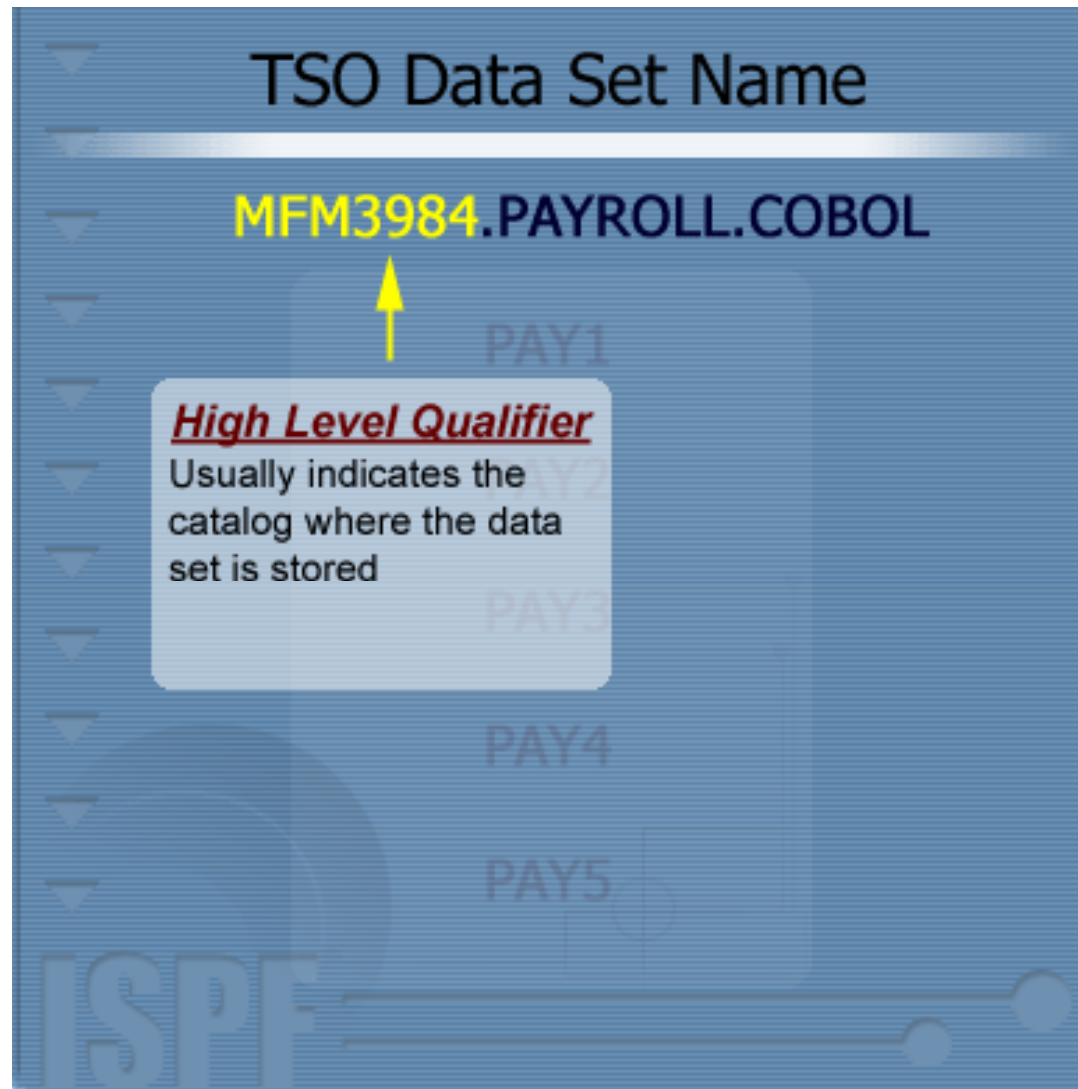
# Why use a PDS



# Naming Conventions



# Naming Conventions - HLQ



# Naming Conventions - User Qualifier






# Naming Conventions - Type of Data

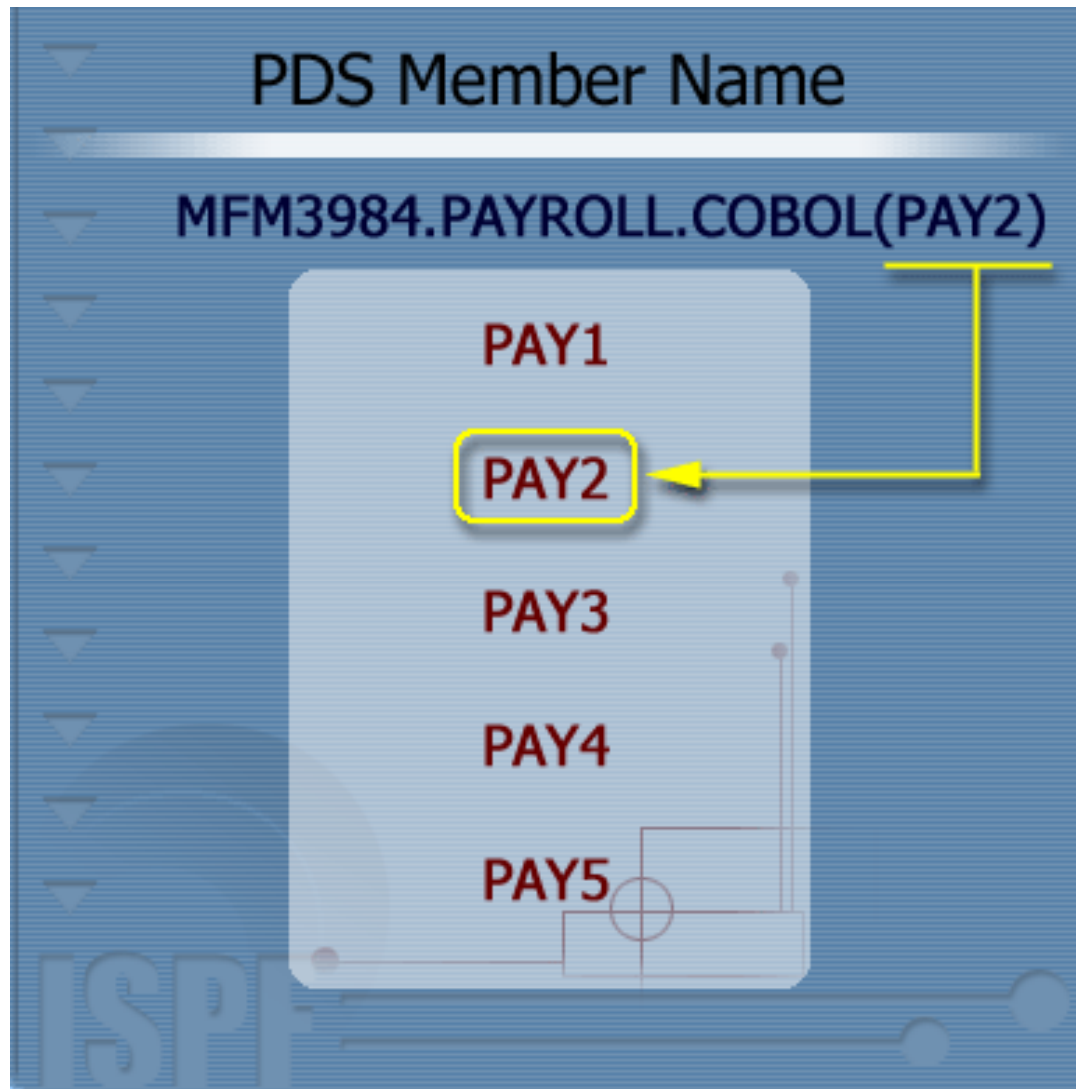
TSO Data Set Name

MFM3984.PAYROLL.**COBOL**



<b><u>ASM</u></b>	Assembler language source code	<b><u>FORT</u></b>	FORTRAN source code
<b><u>CLIST</u></b>	CLIST procedure	<b><u>LOAD</u></b>	Executable program module
<b><u>CNTL</u></b>	JCL job stream used for batch job facility	<b><u>OBJ</u></b>	Object module
<b><u>COBOL</u></b>	COBOL source code	<b><u>PLI</u></b>	PL/I source code
<b><u>DATA</u></b>	Uppercase text data	<b><u>TEXT</u></b>	Upper and lowercase text data
<b><u>EXEC</u></b>	REXX procedure		

# PDS Members



# How data sets are named

- Data set naming convention
  - Unique name (**Maximum 44 characters**)
  - Maximum of 22 name segments: level qualifier
    - The first name in the left: high level qualifier (HLQ)
    - The last name in the right: low level qualifier (LLQ)
    - Level qualifiers are separated by '.'
  - Each level qualifier:
    - From 1 up to 8 characters
    - The first must be alphabetical (A-Z) or special (@ # \$)
    - The 7 remaining: alphabetical, national, numeric (0-9) or hyphen (-)
    - Upper case only
  - Example: **MYID.JCL.FILE2** HLQ: MYID 3 qualifiers

# Data set concepts

- On a mainframe system, files are referred to as *data sets*.
- Each data set has a name that consists of one or more *qualifiers* separated by periods, like MM01.TEST.COBOLE.
- Under ISPF, a data set name consists of a project-id, which is usually your TSO user-id; a library name or group name that can be any name you want; and a type that indicates the kind of data that's stored in the data set.
- You typically store program development data like source, object, and load files in *partitioned data sets*.
- A partitioned data set can contain one or more *members* that are functionally the same as sequential files.
- Before you can use a new or existing data set, you must *allocate* it.

# How data is stored in a z/OS system

- Data is stored on a direct access storage device (DASD), magnetic tape volume, or optical media.
- You can store and retrieve records either directly or sequentially.
- You use DASD volumes for storing data and executable programs, including the operating system itself, and for temporary working storage.
- You can use one DASD volume for many different data sets, and reallocate or reuse space on the volume.

# DASD: Use and terminology

- *Direct Access Storage Device* (DASD) is another name for a disk drive.
- DASD volumes are used for storing data and executable programs.
- Data sets in a z/OS system are organized on DASD volumes.

# EBCDIC

- The IBM S/360 through to the latest zSeries machines use the Extended Binary Coded Decimal Interchange character set for most purposes
- This was developed before ASCII and is also an 8 bit character set
- z/OS Web Server stores ASCII data as most browsers run on PCs which expect ASCII data
- UNICODE is used for JAVA on the latest machines

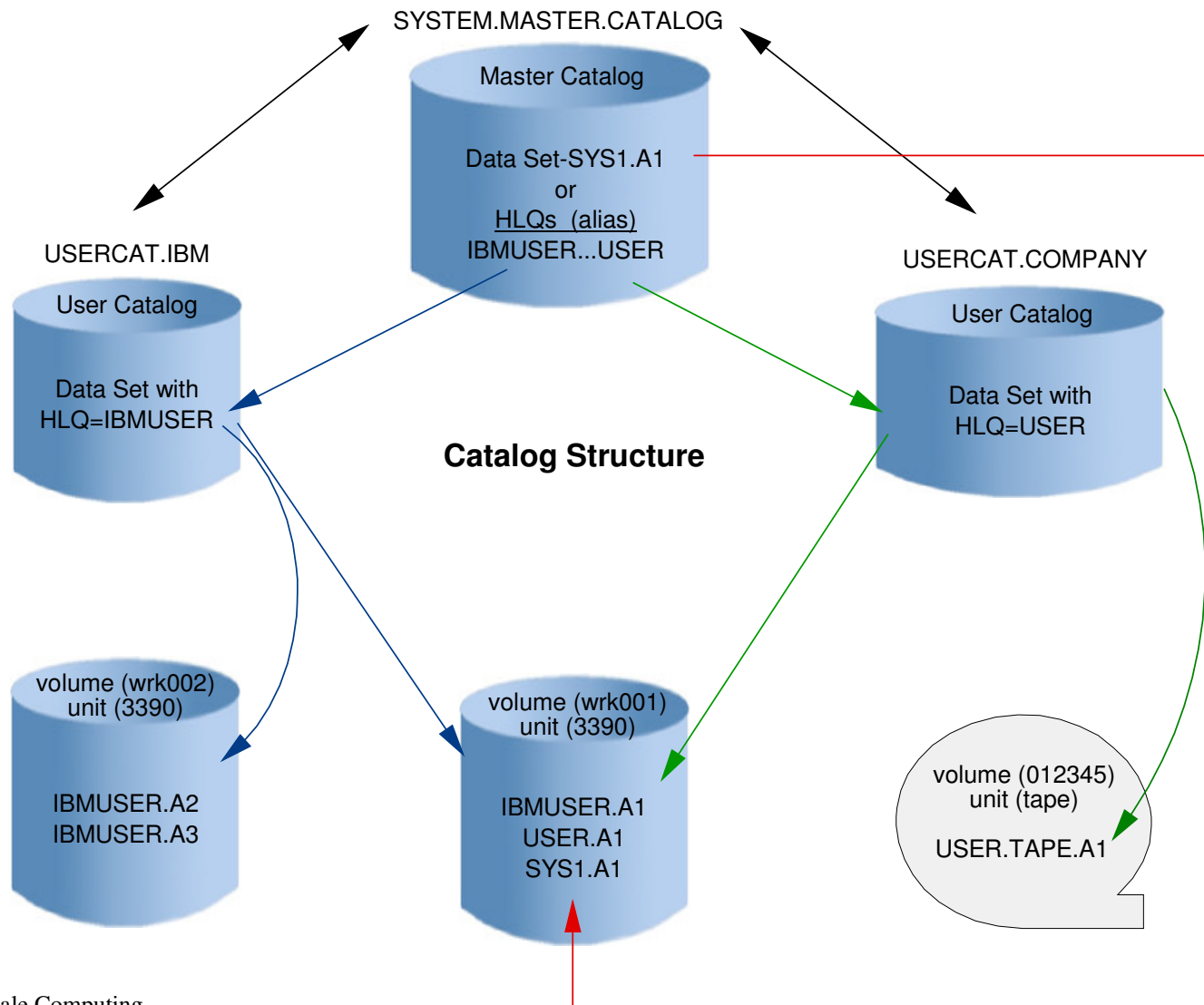
# Catalogs



# How a catalog is used

- A catalog associates a data set with the volume on which the data set is located.
- Locating a data set requires:
  - Data set name
  - Volume name
  - Unit (volume device type)
- Typical z/OS system includes a master catalog and numerous user catalogs.

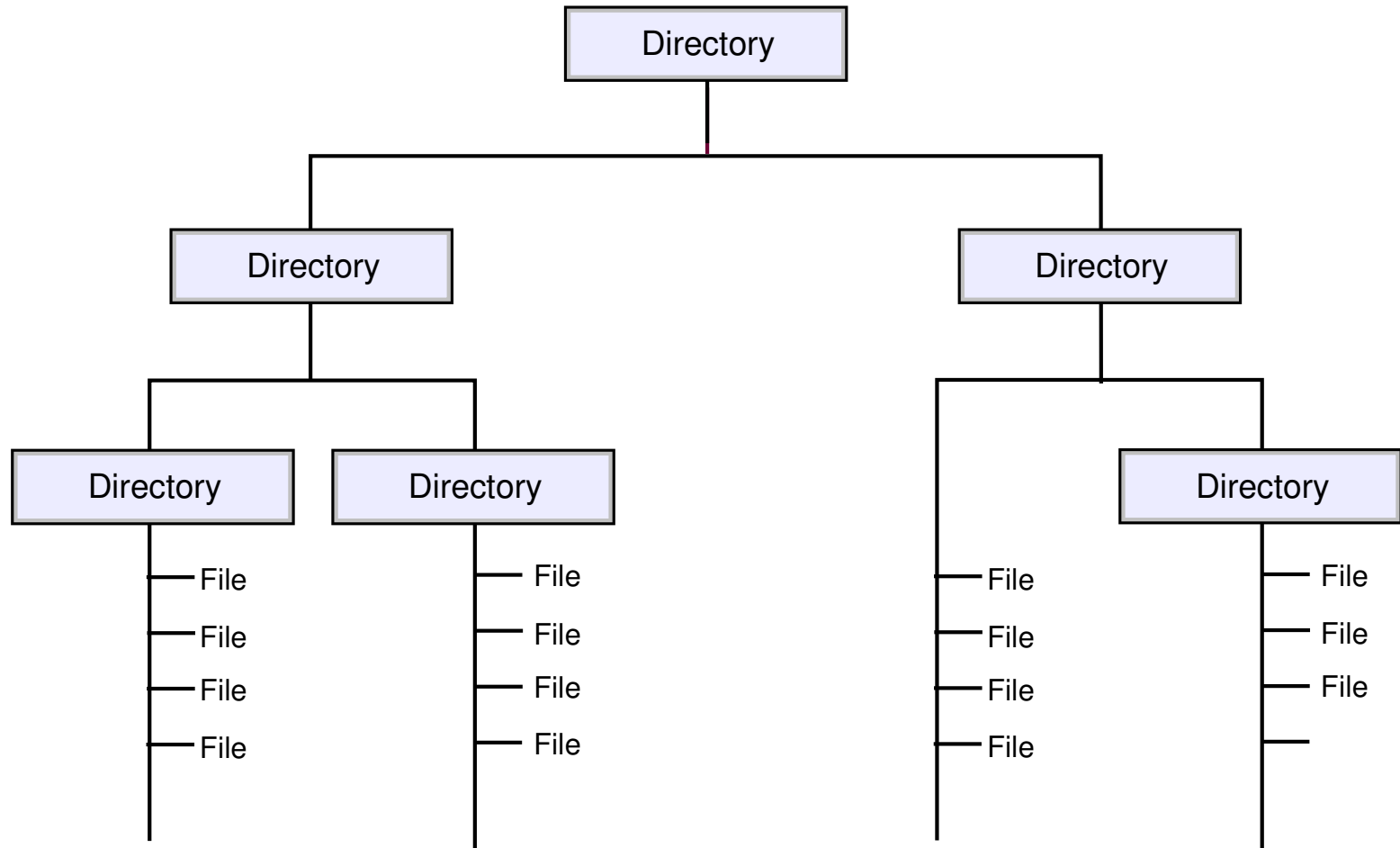
# Catalog Structure



# z/OS UNIX file systems

- z/OS UNIX System Services (z/OS UNIX) allows z/OS to access UNIX files.
- A z/OS UNIX file system is hierarchical and byte-oriented.
- Files in the UNIX file system are sequential files and are accessed as byte streams.
- UNIX files and traditional z/OS data sets can reside on the same DASD volume.

# UNIX file system structure



# TSO

# How do we interact with z/OS?

## **TSO/E**

- Allows users to logon to z/OS and use a limited set of basic commands. This is sometimes called using TSO in its *native mode*.

## **ISPF**

- Provides a menu system for accessing many of the most commonly used z/OS functions.



# TSO overview

- Acronym for Time Sharing Option/Extensions (TSO/E)
- Allows users to create an interactive session with z/OS
- Provides a single-user logon capability and a basic command prompt interface to z/OS
- Most users work with TSO through its menu-driven interface, Interactive System Productivity Facility (ISPF)

# TSO overview (continued)

- In a z/OS system, each user gets a user ID and a password authorized for TSO logon.
- During TSO logon, the system displays the TSO logon screen on the user's 3270 display device or TN3270 emulator.
- z/OS system programmers modify the layout and text of the TSO logon panel to better suit the needs of the system's users.



# TSO/E logon screen

```
----- TSO/E LOGON -----  
  
Enter LOGON parameters below:          RACF LOGON parameters:  
  
Userid   ===> ZPROF  
  
Password ===>  
  
New Password ===>  
  
Procedure ===> IKJACCNT          Group Ident ===>  
  
Acct Nmbr ===> ACCNT#  
  
Size     ===> 860000  
  
Perform  ===>  
  
Command  ===>  
  
Enter an 'S' before each option desired below:  
      -Nomail      -Nonotice      -Reconnect      -OIDcard  
  
PF1/PF13 ==> Help  PF3/PF15 ==> Logoff  PA1 ==> Attention  PA2 ==> Reshow  
You may request specific help information by entering a '?' in any entry field
```

# Lab 1

- Hints
  - Return does not equal “CNRL”
  - If needed, review the tutorial in our Wiki